Low Level Design

HOSPITAL DATASET ANALYSIS

Written By	Aman Rajbhar
Document Version	0.3

DOCUMENT CONTROL

VERSION	DATE	AUTHOR	COMMENTS
0.1	29- AUG - 2024	Aman Rajbhar	Introduction and architecture defined
0.2	30 - AUG - 2024	Aman Rajbhar	Architecture & Architecture description appended and updated.
0.3	31 -AUG - 2024	Aman Rajbhar	Added unit test cases and refined deployment details.

## Contents

1. Introduction

1.1 What is Low-Level design document?

A Low-Level Design (LLD) document provides a detailed and in-depth description of a system's components, modules, and processes. It translates the high-level design into specific, implementable technical instructions. This document typically includes diagrams, data flow descriptions, and specific algorithms or procedures that developers will use to build the system. The LLD is crucial for ensuring that all aspects of the system are thoroughly understood and correctly implemented by the development team.

1.2 Scope

The scope of this LLD document is to detail the architecture and components involved in using SQL for data analysis and visualization in Power BI. It covers the structural aspects of Power BI, including how data is ingested, processed, stored, and visualized. This document is intended for developers, data engineers, and IT professionals involved in setting up and maintaining Power BI infrastructure.
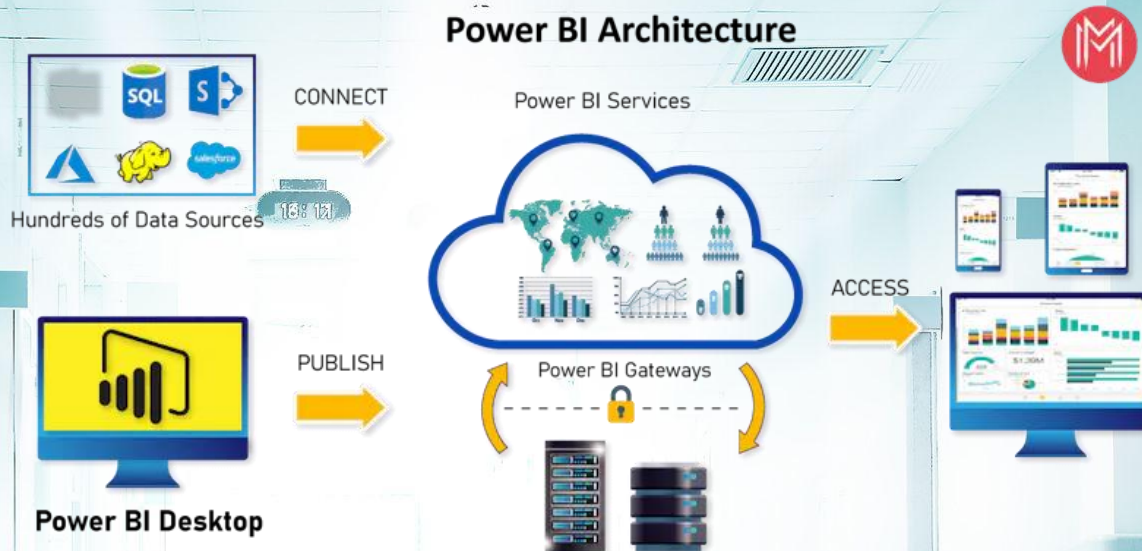
2. Architecture
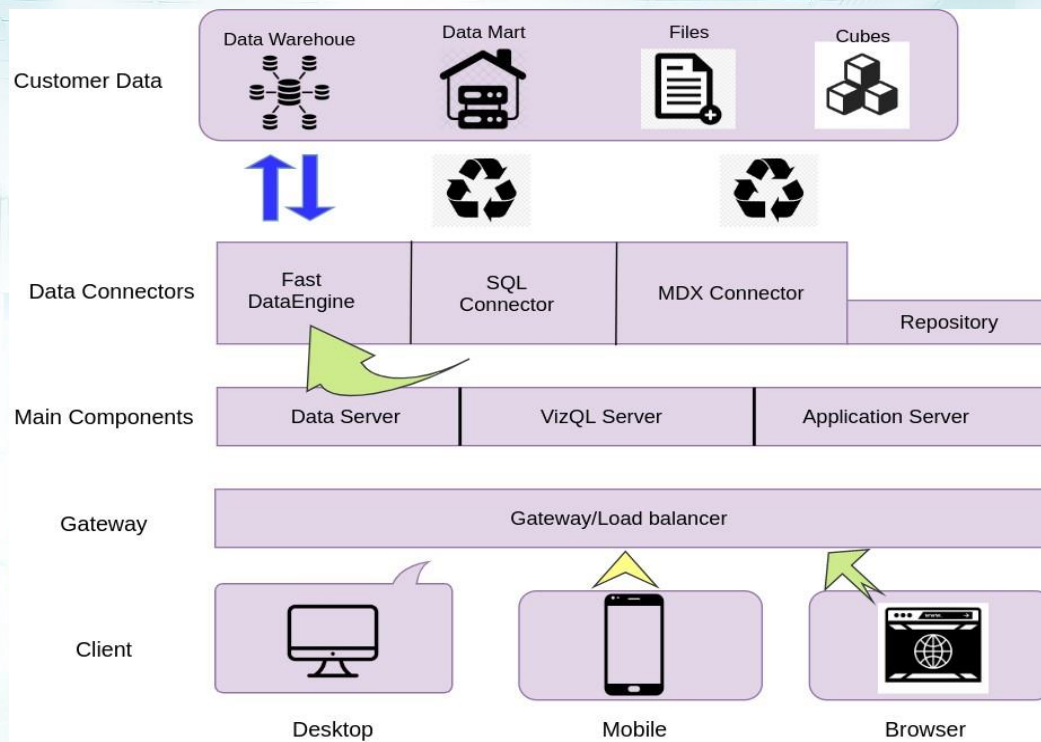
Power BI Server Architecture

Power BI Server Structure is Designed To Support Large-Scale Data Analytics & Visualization tasks. It Consists of Multiple Components That Worked Together To Ensure Efficient Data Processing, Secure Success,and Seamless User Interaction.

The Followig Diagram Shows Power BI Server Structure:-

2.1 Data Sources

- **Variety of Data Sources:** The architecture starts with multiple data sources including SQL databases, SharePoint, Salesforce, Azure, Hadoop, and other cloud services. These are the primary sources from where Power BI pulls the data for analysis and reporting.

- **Data Import:** Data is imported from these sources into Power BI using connectors and APIs that facilitate the extraction, transformation, and loading (ETL) processes.

2.2 Power BI Service

- **Cloud-Based Service:** At the heart of the architecture is the cloud-based Power BI service. This service acts as the central hub where data is processed, modeled, and stored.

- **Data Processing and Analysis:** The Power BI service hosts various components such as the VIZQL Server, Data Engine, and Backgrounder, which handle query execution, data transformation, and scheduling of tasks like data refreshes.

- **Data Storage**: The cloud service also manages the storage of data in secure repositories, ensuring that the data is accessible for analysis while maintaining data integrity and security.

2.3 Data Gateways

- **Gateway/Load Balancer:** The architecture includes gateways that act as a bridge between the Power BI cloud service and on-premises data sources. This allows for secure data transfer between local servers and the cloud environment, facilitating hybrid data environments.

- **Security:** The gateways also ensure that data transfers are secure, utilizing encryption and authentication mechanisms to protect sensitive information.

2.4 Visualization and Reporting

- **Multi-Device Accessibility:** The final layer in the architecture represents the visualization and reporting capabilities of Power BI. Data processed in the cloud is made available through interactive dashboards and reports, which can be accessed on various devices including desktops, tablets, and smartphones.

- **Real-Time Analytics:** The visualizations provide real-time insights, allowing users to interact with the data through filters, slicers, and drill-downs to explore different aspects of the data.

2.5 Power BI Desktop

- **Report Authoring Tool:** Power BI Desktop, shown at the bottom left, is the primary tool for creating reports and dashboards. Users can connect to various data sources, perform data modeling, and design visualizations before publishing them to the Power BI service.

3. Architecture Description

3.1. Data Description

The dataset allows consumers to directly compare across hospitals performance measure information related to heart attack, emergency department care, preventive care, stroke care, and other conditions.

Key	Description
Facility.Name	Name of the hospital
Facility.City	City in which the hospital is located
Facility.State	Two letter capitalized abbreviation of the State in which the hospital is located (e.g., AZ is Arizona)
Facility.Type	Kind of organization operating the hospital: one of Government, Private, Proprietary, Church, or Unknown
Rating.Overall	Overall rating between 1 and 5 stars, with 5 stars being the highest rating; -1 represents no rating.
Rating.Mortality	Above, Same, Below, or Unknown comparison to national hospital mortality
Rating.Safety	Above, Same, Below, or Unknown comparison to national hospital safety
Rating.Readmission	Above, Same, Below, or Unknown comparison to national hospital readmission
Rating.Experience	Above, Same, Below, or Unknown comparison to national hospital patience experience
Rating.Effectiveness	Above, Same, Below, or Unknown comparison to national hospital effectiveness of care
Rating.Timeliness	Above, Same, Below, or Unknown comparison to national hospital timeliness of care
Rating.Imaging	Above, Same, Below, or Unknown comparison to national hospital effective use of imaging
Procedure.Heart Attack.Cost	Average cost of care for heart attacks
Procedure.Heart Attack.Quality	Lower, Average, Worse, or Unknown comparison to national quality of care for heart attacks
Procedure.Heart Attack.Value	Lower, Average, Worse, or Unknown comparison to national cost of care for heart attacks
Procedure.Heart Failure.Cost	Average cost of care for heart failure
Procedure.Heart Failure.Quality	Lower, Average, Worse, or Unknown comparison to national quality of care for heart failures
Procedure.Heart Failure.Value	Lower, Average, Worse, or Unknown comparison to national cost of care for heart failures
Procedure.Pneumonia.Cost	Average cost of care for pneumonia
Procedure.Pneumonia.Quality	Lower, Average, Worse, or Unknown comparison to national quality of care for pneumonia
Procedure.Pneumonia.Value	Lower, Average, Worse, or Unknown comparison to national cost of care for pneumonia
Procedure.Hip Knee.Cost	Average cost of care for hip or knee conditions
Procedure.Hip Knee.Quality	Lower, Average, Worse, or Unknown comparison to national quality of care for hip or knee conditions
Procedure.Hip Knee.Value	Lower, Average, Worse, or Unknown comparison to national cost of care for hip or knee conditions

3.2 Data Collection

The Dataset Was Taken From Provided Link Below:

https://corgis-edu.github.io/corgis/csv/hospitals/

3.3 Data Transformation

In the Transformation Process, we will convert our original datasets with other necessary attributes format.

3.4 Data Insertion into Database

a. Database Creation and connection - Create a database with name passed. If the database is already created, open the connection to the database.

b. Table creation in the database.

c. Insertion of files in the table

3.5 Make the SQL connection and set up the data source

Step 1: Configuring Power BI

- **Launch Power BI Desktop:** Start Power BI Desktop, which is the development tool for creating reports and dashboards.To connect with Power BI you will need to provide information about the server which hosts your database. If you want to connect to a contained database, you can also specify the name of the database.
- **Establish SQL Connection:** In Power BI, to establish a connection to a SQL database, navigate to the **Home** tab and select **Get Data.**
- From the available data sources, choose **SQL Server.**
- In the connection window, input the **server name** and **database** details.
- **Server Name:** The name of the SQL Server.

- **Database Name (Optional):** If you know the database, enter its name; otherwise, leave it blank to browse databases later.

- Select the **Data Connectivity Mode:**
- **Import:** This pulls the data into Power BI's in-memory engine (better for performance but limited by memory).
- **DirectQuery:** Keeps the data in the SQL database and queries it in real-time.
- **Authentication:** Choose the authentication method (Windows, SQL Server, or others), depending on the security configuration of your SQL server.
- Click **OK** to initiate the connection.

Step 2: Configuring Data Source

Data Loading: Once the connection is established, Power BI retrieves metadata from the SQL database. You'll see a list of available tables and views. Select the necessary tables to import or query the data you need for analysis.

Data Preview: Power BI provides a preview of the selected data. Here you can filter rows or columns, choose specific fields, and even apply basic transformations.
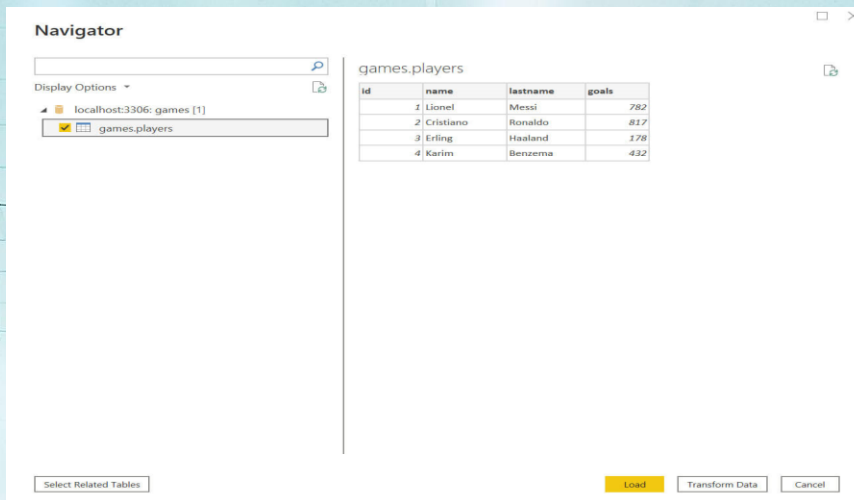
Data Transformation: After selecting the data, Power BI automatically loads it into the **Power Query Editor** (if using the Import mode).

In the Power Query Editor, you can clean and transform the data by filtering rows, changing data types, renaming columns, or merging tables, among other operations.

After making the necessary changes, apply the transformations by clicking **Close & Apply**. The data will be imported or queried directly from SQL and made available for report-building.
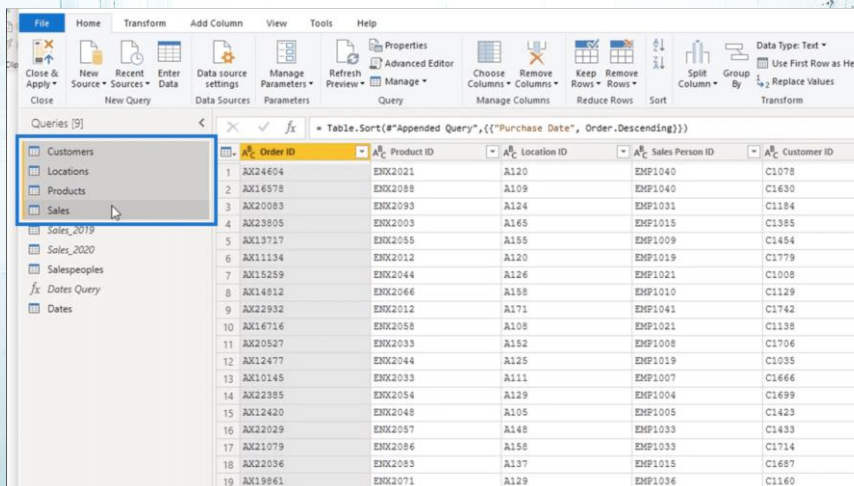
## 3.6 Export Data from Database

Direct Export from SQL:

- If your analysis or visualization requires exporting data from SQL, Power BI does not directly export data from the database itself. However, there are indirect ways:

  - **Export Table/View Data:** You can export data from Power BI visuals or tables to Excel or CSV files. To do this:

    1. **Create a Visual/Table:** Generate a table visualization in Power BI with the relevant data from SQL.

    2. **Export Data:** Right-click on the visual and select **Export Data**. Power BI will allow you to save the table's content as a CSV or Excel file. This is ideal for small datasets or summary tables.

  - **Advanced Data Export:** For more sophisticated needs, such as exporting specific results from SQL queries, you may have to export the data from the SQL Server directly (using SQL Server Management Studio or scripts) or use another tool alongside Power BI to extract this data.

3.7 Deployment

Once the report is ready, I need to publish it to **Power BI Service** so it can be shared and consumed by others.

1. **Sign in to Power BI Service:** I go to the Home tab in Power BI Desktop and click "Publish," which prompts me to sign in with my Power BI account.
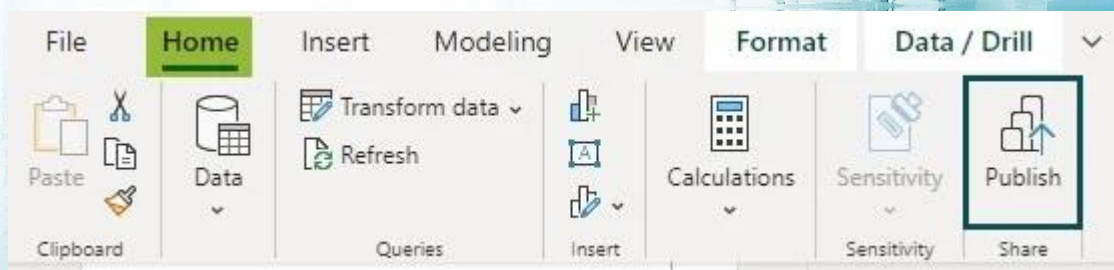
2. **Choose a Workspace:** After signing in, I select a workspace in Power BI Service where I want to publish the report. This workspace is where I'll store reports, dashboards, and datasets for collaboration.
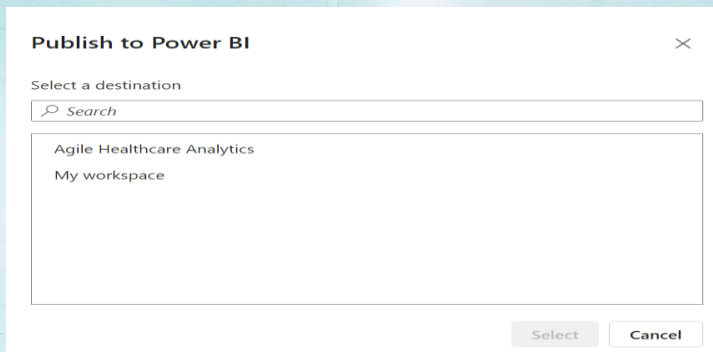
3. **Publish the Report:** I click "Publish" to upload my report. In a few moments, Power BI Service confirms that the report has been successfully published.
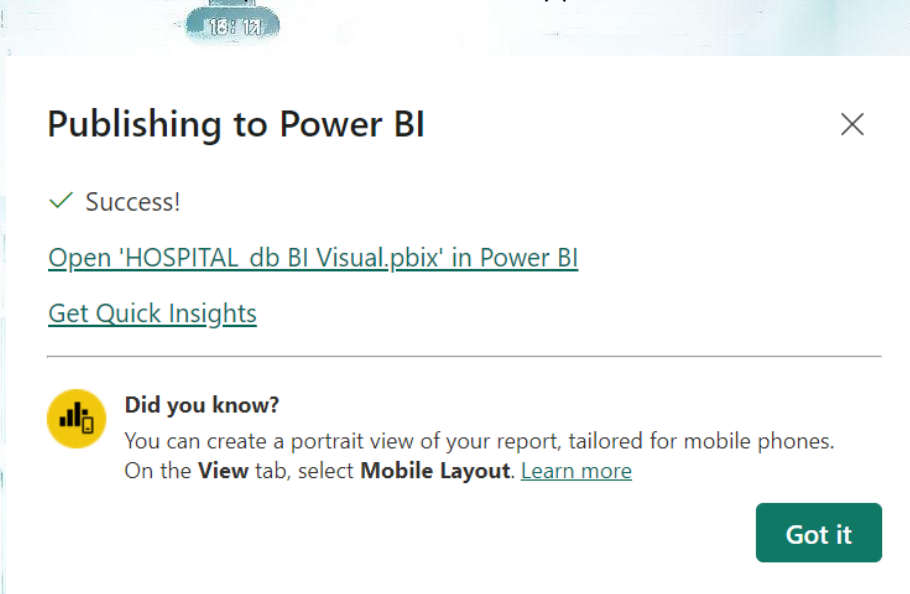

Once published, I set up a **Data Refresh Schedule** if my report uses the Import method. I configure the frequency of these updates (daily, weekly, etc.) to ensure my report stays up-to-date.

Security and Access Control: At this point, I ensure only authorized users have access to the report. Power BI integrates with **Azure Active Directory**, which allows me to manage permissions securely based on user roles in the organization.

Report Sharing: With the report deployed, I can share it with specific individuals or teams, embed it in applications like SharePoint or Microsoft Teams, or provide users with interactive links.

Key Considerations: Throughout this process, I keep an eye on performance, making sure I minimize the data pulled from SQL Server by aggregating data and optimizing queries. If my report has a large audience or involves big datasets, I might need to consider **Power BI Premium** to ensure performance remains smooth for everyone accessing it.

This end-to-end process ensures my Power BI project is properly connected to SQL, configured for analysis, and efficiently deployed for real-world use.
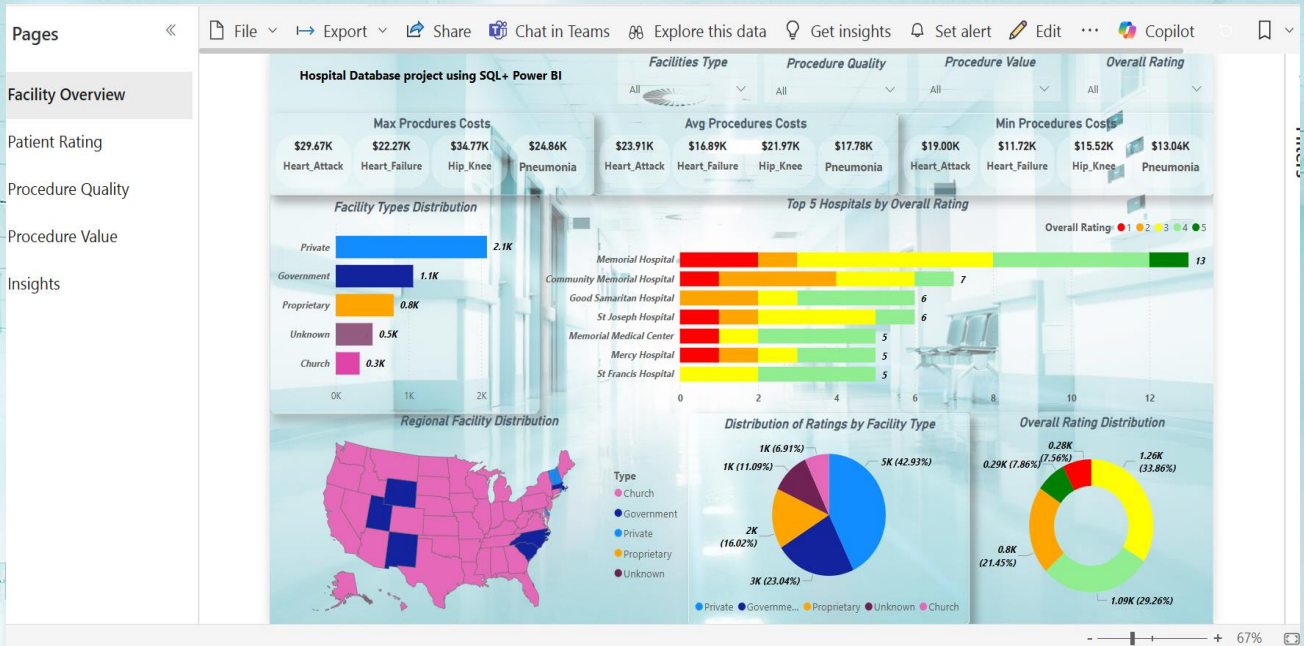
4. Unit Test Cases

TEST CASE DESCRIPTION	EXPECTED RESULTS
Facility Type slicer	When clicked on the slicer, a dropdown should occur which has various parameters of the Facility types.
Overall Rating Parameter	When clicked on the slicer, a dropdown should occur which describes the parameters of the Overall Rating.
Relation Between Overall Rating and Facilities Types	Here a Pie Chart is shown of Overall Rating Distribution by Facilities Types.
Facilities Types Distribution	The Visual bar chart is showing Distribution of Facilities Types.
Map showing Facility Distribution	The visual should show a Map of Regional Facility Disrbuition in Diferent Countries.
Min, Max & Avg. of All Procedure Cost	This is an important KPIs visual of Min, Max & Avg. of All Procedure Cost.
Procedure Quality slicer	When clicked on the slicer, a dropdown should occur which has various parameters of the Procedure Quality.
Procedure Value slicer	When clicked on the slicer, a dropdown should occur which has various parameters of the Procedure Value.